Saken Kenzhegulov
20132027

MP#2: Report

myheader.h:

There I declare packet structure as given in the project description.

Client.c:

After making all connections, there I declare two pthreads: ‘thread1’ – for sending packets to server, ‘thread2’ – for receiving packets from server. Accordingly, on each pthreads I run two functions: ‘send’ and ‘receive’, respectively. Additionally, I created ‘args’ struct to pass arguments to these functions.

In ‘send’ function I generate given ‘npkts’ of random packets and arrival times according to the given mean of packet size and arrival time. For this I used created function ‘exp_dis’ which calculates random according to the given mean of exponentially distributed function. At the end, I create one more packet with ‘seqno=-1’ to notify server that all packets were sent and received.

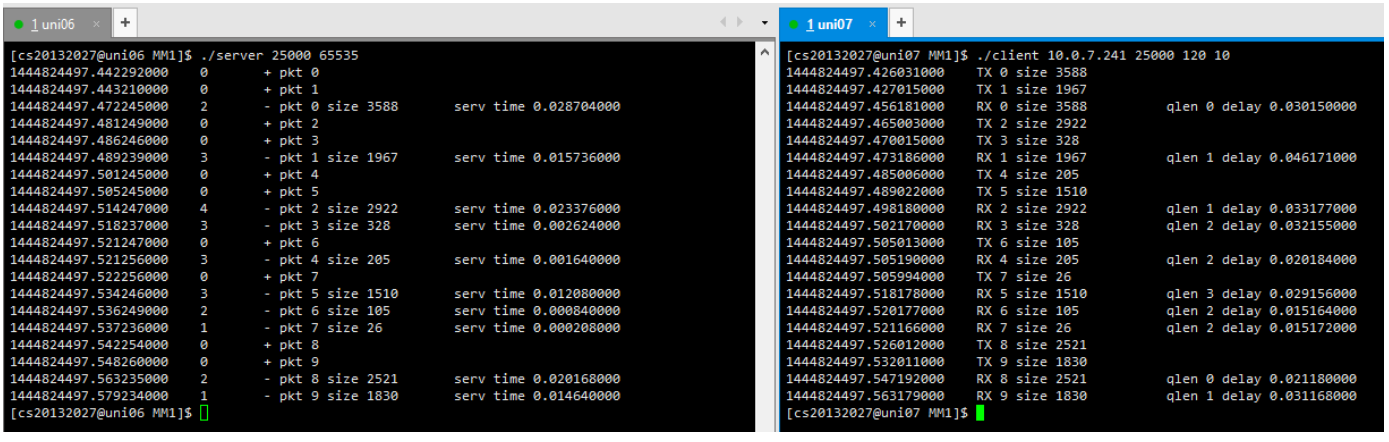
In ‘receive’ function I receive all packets from server, and calculate delay time. When I get packet with ‘seqno’ equal to ‘nkpts-1’, which means it is the last packet, I stop receiving.

Server.c:

I created FIFO queue structure with all needed functions and variables.

After creating socket and starting server, start to receive packets. There is a child pthread function ‘service’ to work on packets and send it back to the client. ‘Service’ is called only when queue is not empty and other pthread is not working with queue already.

In ‘service’, proceed all packets in queue while queue is not empty. After getting a packet, calculate a service time and simulate delay. Then send this packet back to client, now with updated value of ‘serverQueueLen’. Whenever queue becomes empty, thread stops its work.



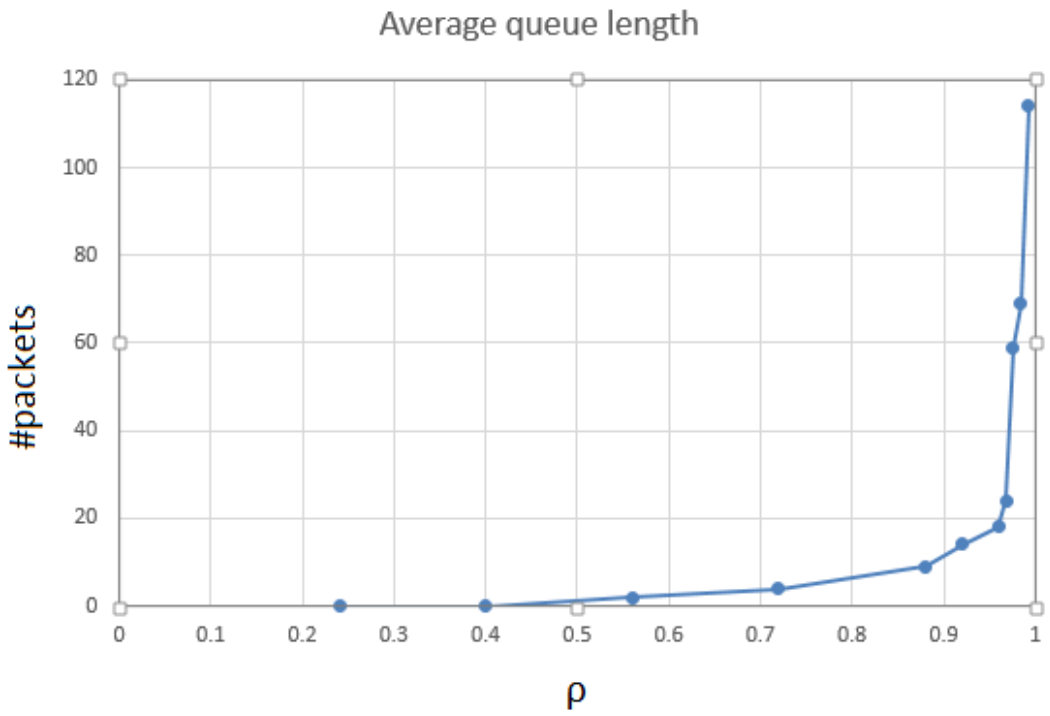


Figure 1

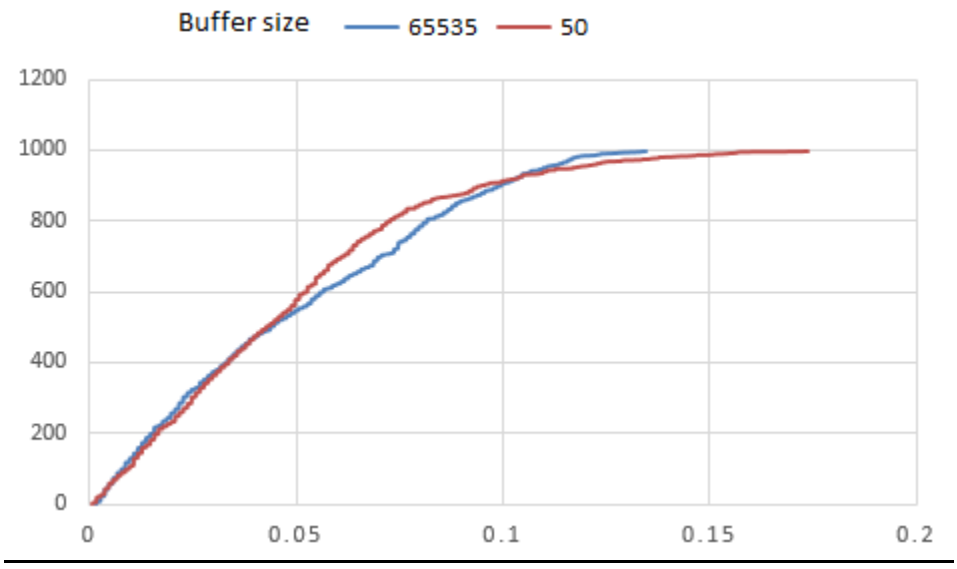


Figure 2